Multiview Convolutional Neural Networks for Multidocument Extractive Summarization

Yong Zhang, *Student Member, IEEE*, Meng Joo Er, *Senior Member, IEEE*, Rui Zhao, and Mahardhika Pratama, *Member, IEEE*

***Abstract*—Multidocument summarization has gained popularity in many real world applications because vital information can be extracted within a short time. Extractive summarization aims to generate a summary of a document or a set of documents by ranking sentences and the ranking results rely heavily on the quality of sentence features. However, almost all previous algorithms require hand-crafted features for sentence representation. In this paper, we leverage on word embedding to represent sentences so as to avoid the intensive labor in feature engineering. An enhanced convolutional neural networks (CNNs) termed multiview CNNs is successfully developed to obtain the features of sentences and rank sentences jointly. Multiview learning is incorporated into the model to greatly enhance the learning capability of original CNN. We evaluate the generic summarization performance of our proposed method on five Document Understanding Conference datasets. The proposed system outperforms the state-of-the-art approaches and the improvement is statistically significant shown by paired *t*-test.**

***Index Terms*—Convolutional neural networks (CNNs), deep learning, multidocument summarization (MDS), multiview learning, word embedding.**

I. Introduction

AUTOMATIC text summarization has been widely researched in recent years with the explosive growth of accessible information due to the rapid development of Internet and computing technology. Multidocument summarization (MDS) can help users to sift through vast volumes of information, and to quickly identify the most vital information. There are two categories of MDS, namely, *abstractive summarization* and *extractive summarization*. The abstractive summarization methods involve sentence compression and reformulation which resemble human summarizers, but the linguistic processing procedure is so complicated that it is very difficult to be implemented [1]. On the other hand, the extractive methods directly extract the most informative sentences in a document to form the final summary. Due to its simplicity, most works done in this area fall in the category of extractive summarization.

Extractive summarization methods can be roughly classified into three types [2]: 1) methods based on sentence positions and article structure; 2) unsupervised methods; and 3) supervised methods. For the first category, important sentences such as those in the introductory or concluding part, will be selected to fit into the summary. On the other hand, unsupervised methods rank sentences by salience scores which are estimated based on statistical and linguistic features and extract the top ones to constitute the summary. In contrast to unsupervised methods, supervised approaches utilize a set of training documents together with their corresponding hand-crafted summaries to train a binary classifier which is used to predict whether a sentence should be included in the summary or not. Our new method belongs to the supervised category. The existing methods have some obvious drawbacks which can be summarized as follows.

1) Most of the existing methods depend on hand-crafted features to represent sentences which result in intensive human labor.
2) They can hardly capture semantic and syntactic information contained in documents simultaneously.
3) Their summarization capabilities are not good enough.

This paper aims to solve the problem of designing laborious hand-crafted features and enhance summarization capabilities. We leverage on the word embedding technique to represent sentences and propose an easy-to-implement system. We are among the earliest research groups using word embedding to reduce the labor of feature engineering in the research area of document summarization. Both [3] and [4] employed convolutional neural networks (CNNs) and pretrained word embedding to summarize documents. However, their works still use several hand-crafted features together with the word vectors. Our model offers a new choice and only uses another very simple and easy-to-implement sentence position feature. The experiment results demonstrate that the new method improves summarization capability compared with existing methods. Based on the idea of word embedding, an innovative sentence position embedding technique is developed to represent the sentence position feature and further enhance the learning capacity of our proposed system.

In recent years, deep learning methods together with pretrained word embedding have achieved remarkable results for various natural language processing (NLP) tasks. Deep learning models reduce the intensive labor of feature engineering

Manuscript received July 15, 2016; accepted October 29, 2016. This paper was recommended by Associate Editor Y. Zhang.

Y. Zhang, M. J. Er, and R. Zhao are with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 636798 (e-mail: yzhang067@e.ntu.edu.sg; emjer@ntu.edu.sg; rzhao001@e.ntu.edu.sg).

M. Pratama is with the Department of Computer Science and IT, La Trobe university, Melbourne, VIC 3086, Australia (e-mail: m.pratama@latrobe.edu.au).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2016.2628402

because they can learn features from data automatically. Deep learning takes advantage of the increase in the amount of available computation and data and produces extremely marvelous results in speech recognition, visual object recognition, object detection, and many other complicated tasks [5]–[7]. More detailed information and recent developments in deep learning can be found in the review paper published in *Nature* [8].

Word embedding is a technique representing words with low-dimension vectors, solving the problem of curse of dimension and sparsity of conventional bag-of-words method [9], [10]. Word embedding has drawn great attention in recent years because it can capture both semantic and syntactic information. The idea of word embedding has a very long history but has become popular since Bengio *et al.*'s works [9], [11] in which each word was represented by a vector and the concatenation of several previous word vectors is employed to predict the next word using a neural networks language model. Since then, a lot of researchers have explored the salient features of vector representation of words using neural networks [10], [12]–[15]. The most popular architectures of neural networks language models were proposed by Mikolov *et al.* [10], [15]. They developed their methods in the context of recurrent neural networks and proposed two efficient word representation estimation models, namely, continuous bag-of-words (CBOW) and skip-gram. The proposed models can map the words into a vector space, where words with similar semantics lie close to each other, for example, vec(American) is closer to vec(France) than to vec(Bread). The vector representations of words can even preserve the semantic relationship. An example is given as follows:

$$\mathrm{vec}(\mathrm{France}) - \mathrm{vec}(\mathrm{Paris}) = \mathrm{vec}(\mathrm{China}) - \mathrm{vec}(\mathrm{Beijing}). \quad (1)$$

This suggests that vector representations capture the fact that *Paris* and *Beijing* are capitals of *France* and *China*, respectively. Word embedding has been applied to many NLP applications, such as name entity recognition [12], word sense disambiguation [14], parsing [16], tagging [17], and machine translation [18].

CNNs was originally proposed for computer vision by LeCun *et al.* [19]. It produces excellent results for computer vision tasks [20]–[22] and recently has been proven to be effective for various NLP tasks as well, such as part of speech tagging [23], sentence modeling [24], [25], semantic embedding [26], and sentence classification [27], to name a few. As CNN has demonstrated great power in latent feature presentation [28], [29], we propose an enhanced CNN model termed multiview CNNs (MV-CNNs), to obtain the features of sentences and rank sentences jointly, and to build a novel summarization system. Our new approach is developed based on the basic CNN but incorporates the idea of multiview learning to enhance the learning capability. The new model leverages on pretrained word embedding to refrain people from intensive feature engineering labor. Furthermore, the word embedding can efficiently help to improve the learning capability of the proposed model.

The basic CNN structure in this paper adopts two consecutive convolution layers followed by a max-pooling layer to express sentence vectors. One problem of representing sentences is that users' opinions toward sentences may vary tremendously. People tend to summarize documents very differently due to their perceptions and biases as well as different background knowledge and intentions. In order to address the issue, we incorporate the idea of multiview learning. Multiview learning is a paradigm designed for problems, where data come from diverse sources or different feature subsets, also known as, multiple views. It employs one function to model a particular view and jointly optimizes all the functions to exploit the redundant views of the same input data and improve the learning performance [30]. Multiview learning follows two significant principles, namely, complementary and consensus principles. The complementary principle employs complementary information underlying multiview data to comprehensively and accurately describe the data. The consensus principle aims at maximizing the agreement of distinct learners trained on multiple views. The new model follows a two-level complementary principle and the consensus principle. Our model theoretically and experimentally proves that multiview is a plausible direction to improve CNN learning capability.

In the proposed model, multiple convolution filters with varying window sizes are used to complete the sentence representation and sentence ranking tasks. This technique has been demonstrated to be very useful for computer vision and NLP tasks. We find the technique conforms to the spirit of multiview learning to some extent. The CNNs with different filter window sizes can be regarded as different summarizers. The saliency scores of sentences assigned by different CNNs will be combined appropriately to obtain the final scores, just like that different summarizers negotiate with each other in order to determine the final summaries after getting their summaries independently. Furthermore, we not only combine the final scores rated by each distinct-filter-size CNN, but also concatenate sentence representations acquired by each CNN before calculating the saliency scores of sentences and use the new sentence representations to analyze the sentences. This is analogous to the collaborative process that different summarizers exchange one another's opinions first and perform the summarization task together. The above procedure forms a two-stage complementary principle. The consensus principle is also used to maximumly exploiting the power of multiview learning. Consensus principle can be regarded as that different summarizers discuss and minimize the discrepancy of their opinions while the complementary principle can be thought of as overcoming the problem that each summarizer cannot take everything into consideration. Another innovative component, namely, sentence position embedding is also used to improve learning ability. We evaluate our new model on five Document Understanding Conference (DUC) generic summarization benchmark datasets using Recall-Oriented Understudy for Gisting Evaluation (ROUGE) metric. The proposed method achieves superior performance on all the measure metrics on the five datasets compared with state-of-the-art methods. MV-CNN obtains 1.28% higher Rouge-1 score, 4.35% higher Rouge-2 score, and 3.68% higher Rouge-SU4 score relatively compared with the best existing summarization system on the DUC2006 dataset. The main contributions of this paper are summarized as follows.

1) Multiview learning is applied to CNN to enhance the learning capability of the original CNN. The two-level complementary and consensus principles are fully exploited to improve the model's learning performance. To the best of our knowledge, this is the first time that the combination of multiview learning and CNN is used for MDS.
2) With the help of pretrained word embedding, human feature engineering is not needed. This makes our model much easier to be implemented.
3) Sentence position embedding is incorporated into the model to advance the learning capacity of the proposed model to a higher level.

This paper is organized as follows. Section II gives a brief review of related works. In Section III, the proposed model is described in details. The performance of the proposed method is compared with the state-of-the-art methods in Section IV. Conclusions are drawn in Section V.

II. Related Works

The proposed model applies deep learning to extractive MDS and exploits multiview learning to enhance the learning performance. The existing extractive summarization techniques are reviewed first. And the two most related methods, namely, deep learning in MDS and multiview learning, are briefly introduced in this section as well.

A. Extractive Summarization

As foreshadowed in Section I, extractive summarization methods can be roughly classified into three types [2], [31]: 1) methods based on sentence positions and article structure; 2) unsupervised methods; and 3) supervised methods. The famous LEAD is a representative of the first category, which simply extracts the leading sentences to summarize a document [32]. However, the first kind of methods can only be applied to strictly structured documents like newswire articles. In contrast, unsupervised methods can be applied to wider range of document formats and demonstrate stronger summarization ability. Latent semantic analysis (LSA) [33] is a well-known unsupervised method applying the singular value decomposition (SVD) on the term frequency matrix and then selecting sentences with top eigenvalues. Graph-based ranking methods play a significant role in unsupervised MDS in recent years, such as the LexRank [34] and the TextRank [35]. They first build a graph of sentence similarities and then calculate the importance of a sentence by inspecting its links to all other sentences in the graph recursively. Some other unsupervised extractive MDS methods, include the maximal marginal relevance [36], the Markov random walk [37], and the submodularity-based methods [38]. Supervised methods have also been widely researched. In [39], a probabilistic approach termed conditional random field was proposed to rank full sentences while researchers in [40] and [41] trained their models based on n-gram regression. Recently, some researchers measured the salience of both sentences and n-grams [42], [43]. Hu and Wan [42] took advantage of support vector regression and the approach in [43] was developed based on the recursive neural networks. Our new method belongs to the supervised category.

B. Deep Learning in Multidocument Summarization

Many recent works have already been done using deep learning methods to summarize documents. Restricted Boltzmann machine (RBM) was incorporated to generate generic summary in [44]. However, it only extracted four linguistic features per sentence as the input to the RBM and its performance was not very satisfactory. Cao *et al.* [43] ranked the sentences with recursive neural networks, obtaining much better performance compared with the traditional methods. However, it used hand-crafted features to represent the input of the model. Besides, the RNN is built via a tree structure and its performance heavily depends on the construction of the textual tree. The tree construction can be very time consuming. One simple method termed paragraph vector [45] was developed to encode sentences into vector representations. The model was developed based on the architecture of CBOW and skip-gram, but proved to perform suboptimally on some tasks. Denil *et al.* [3] attempted to summarize documents with hierarchical CNNs. They employed CNNs to extract sentence features from words and another CNN on top of the learned sentence features to obtain document features. Deconvolutional networks were used to train their models but a lot of hyper-parameters had to be tuned [46]. Our new model is most similar to the model proposed by Cao *et al.* [4]. Cao *et al.* [4] also concatenated CNNs with distinct window sizes to obtain sentence features and rank sentences. Several hand-crafted features were introduced and concatenated with sentence features for sentence regression. The main difference between our model and Cao *et al.*'s [4] model lies in the incorporation of the idea of multiview learning which enables our model to achieve superior performance. Actually, their model's concatenation of CNNs with distinct window size forms the first-level complementary principle of our model, but our model is developed based on two-level complementary principle and consensus principle.

C. Multiview Learning

Multiview learning is a paradigm designed for problems, where data come from diverse sources or feature subsets, also known as, multiple views. It employs one function to model a particular view and jointly optimizes all the functions to exploit the redundant views of the same input data and improve the learning performance [30]. Multiview learning has been widely researched in recent years and can be mainly divided into three categories.

1) Co-training algorithms which alternately train two separate learners using features from distinct views to maximize the mutual agreement on the prediction of the two classifiers on the labeled dataset as well as to minimize the disagreement on the prediction on the unlabeled

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

dataset [47], [48]. These approaches combine multiple views at a later stage after training the base learners.

2) Multiple kernel learning algorithms which exploit separate kernels that naturally correspond to different views and combine kernels to integrate multiple information sources [49], [50]. Views (kernels) are combined at an intermediate stage before or during the training of learners.
3) Subspace learning-based approaches which aim at obtaining an appropriate subspace shared by multiple views by assuming that input views are generated from a latent subspace [51], [52]. These approached can be regarded as prior combinations of multiple views.

Recently, there is an attempt to combine multiview learning and CNN in the research area of computer vision. Su *et al.* [53] employed multiple CNNs to compile information from a collection of 2-D views of 3-D objects into a compact shape descriptor of the object which offers better recognition performance compared with single-view methods. This method regards different 2-D shapes extracted from 3-D objects as distinctive views. However, this method used multiview learning at a very superficial level and did not consider the inner connections of different views of the observed objects. Lin *et al.* [54] employed CNN to deal with multiple sources of microblog data in order to detect psychological stress. However, they handled the data separately and did not consider their inner connections as well. In contrast, our model aims at digging underlying distinct views of sentences by using different CNNs. It is like a multiple-kernel-learning method where each CNN can be seen as one kernel.

III. Proposed Model

Extractive summarization is defined as the selection process of salient sentences in a corpus of documents so as to generate a brief summary which can best describe the subject. We denote the document corpus as $C = \{D_1, D_2, \ldots\}$, in which D_i is the ith document in the corpus. Each document consists of a set of sentences. We include all the sentences in the corpus to constitute the candidate set $\text{CS} = \{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_N\}$, where $\mathbf{s}_i$ is the vector representation of the ith sentence in the corpus. The term N is the number of distinct sentences in the corpus. The generation of sentence representation is a nontrivial task. Most conventional methods require hand-crafted features which result in intensive human labor. Our proposed method leverages on the innovative CNN model as well as word embedding to obtain sentence representations. The model will assign saliency scores to all the sentences. After assignment of salience scores, the sentences can be ranked so that the sentences in the candidate set with high scores will be selected as summary sentences. The selected sentences form the summary set $\text{SS} = \{\mathbf{s}_1^*, \mathbf{s}_2^*, \ldots, \mathbf{s}_L^*\}$. Note that $L << N$ and $\text{SS} \subset \text{CS}$.

This section will first introduce the structure of the basic CNN of our model and give detailed information of the proposed MV-CNN model. The entire summarization procedure using the new approach is given.

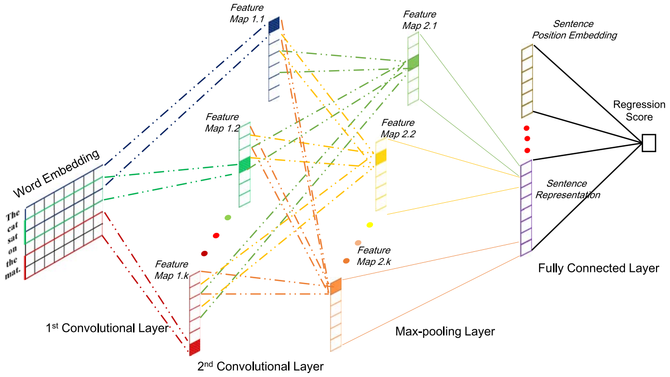


Fig. 1. CNN architecture.

A. Convolutional Neural Networks

In this section, we first give the structure of basic CNN used in our model. In [27], it was shown that a simple CNN model with one convolution layer followed by one max-pooling layer is able to perform extremely well for sentence classification. The CNN approach demonstrates excellent efficiency in latent feature presentation and can naturally address variable-length sentences. We add one more convolution layer before pooling so as to extract deeper and more abstract information contained in the sentences. The previous classification model is adapted to solve a regression task in order to rank sentences. As shown in Fig. 1, this specific convolutional architecture only consists of two convolution layers followed by a max-pooling layer. Finally, the sentence representation vector obtained by the CNN, concatenated with sentence position embedding, is fed to the fully connected layer to calculate the regression score of the input sentence.

1) Convolutional Layer: Convolutional layers play critical roles in the success of the CNNs because they can encode significant information about input data with significantly fewer parameters than other deep learning architectures. Empirical experiences in the area of computer vision suggest that deep architectures with multiple convolutional layers are necessary to achieve good performance [8]. However, only one convolutional layer was used to achieve satisfactory performance in the sentence classification task in [27]. This may be due to the fact that the datasets used by him are not very large. Our experiment results show that two convolutional layers are needed to achieve sufficient model capacity. The convolution operation in our model is conducted in 1-D. The first convolutional layer is done between k filters $\mathbf{W}_1 \in \mathbb{R}^{md \times k}$ and a concatenation vector $\mathbf{x}_{i:i+m-1}$ which represents a window of m words starting from the ith word, obtaining features for the window of words in the corresponding feature maps. The term d is the dimension of word embedding. A word vector is defined as follows:

$$\mathbf{x} = Lw \tag{2}$$

where $w \in \mathbb{R}^{|V|}$ is a one-hot vector, where the position corresponding to the word is one while the other positions are zeros, and $L \in \mathbb{R}^{d \times |V|}$ is a word-representation matrix, and $|V|$ is the vocabulary size. We can easily adopt off-the-shelf word embedding matrices that are already on line.

Multiple filters with different initialized weights are used to improve the model's learning capacity. The number of filters k is determined using cross validation. The convolution operation is governed by

$$\mathbf{c}_i^1 = g(\mathbf{W}_1^T \mathbf{x}_{i:i+m-1} + \mathbf{b}_1) \in \mathbb{R}^k \tag{3}$$

where $x_i \in \mathbb{R}^d$. The term $\mathbf{b}_1$ is a bias vector and $g(\cdot)$ is a nonlinear activation function, such as sigmoid, hyperbolic tangent, or rectified linear units (ReLUs). The ReLU has become a standard nonlinear activation function of CNN recently because it can improve the learning dynamics of the networks and significantly reduce the number of iterations required for convergence in deep networks. We employ a special version of the ReLU called LeakyReLU [55] that allows a small gradient when the unit is not active. It helps further improve the learning performance compared with ReLU.

Suppose the length of a sentence is n. As the word window slides, the feature maps of first convolutional layer can be represented as follows:

$$\mathbf{c}^1 = \left[\mathbf{c}_1^1, \mathbf{c}_2^1, \ldots, \mathbf{c}_{n-m+1}^1\right]. \tag{4}$$

We set the window size and number of feature maps of the second convolutional layer the same as the first convolutional layer for easy understanding because we distinguish the distinct CNNs by their window sizes. Therefore, the output of the second convolutional layer is given by

$$\mathbf{c}_j^2 = g\left(\mathbf{W}_2^T \mathbf{c}_{j:j+m-1}^1 + \mathbf{b}_2\right) \in \mathbb{R}^k \tag{5}$$

where $\mathbf{W}_2 \in \mathbb{R}^{mk \times k}$ and $g(\cdot)$ is also the nonlinear activation function LeakyReLU.

As the window slides on the previous convolutional layer, the feature maps of the second convolutional layer are given as follows:

$$\mathbf{c}^2 = \left[\mathbf{c}_1^2, \mathbf{c}_2^2, \ldots, \mathbf{c}_{n-(m-1)*2}^2\right]. \tag{6}$$

2) Max-Pooling Layer: Max-pooling layers are useful in reducing the number of parameters in the network by reducing the spatial size of the vector representation. We conduct a max-pooling operation: $\mathbf{h} = \max(\mathbf{c}^2) \in \mathbb{R}^k$ to obtain features corresponding to the filters. The idea behind the max-pooling operation is that it can reduce feature dimensionality and lead to a fixed-length feature vector regardless of variable sentence lengths.

3) Fully-Connected Layer: After the max-pooling layer, we obtain the penultimate layer $\mathbf{h} = [h_1, \ldots, h_k]^T$, k is the number of filters, which is the vector representation of the input sentence. Besides the sentence representation vector, we also feed *sentence position embedding* into the fully connected layer. The location where a sentence occurred in one document can be critical for determining how important the sentence is for the document. We use position embedding instead of a position integer so that the position feature will not be drowned by the long sentence vector. In [29], word position embedding was proposed to facilitate the relation classification task. Inspired by their ideas, we incorporate sentence position embedding into our proposed CNN model. Generally speaking, sentences appearing in the beginning part or conclusion section of one document may contain the most useful information. Therefore, we denote the position of a sentence s_i according to the following equation:

$$p(s_i) = \begin{cases} 0 & s_i \in S_{1:3} \\ 1 & s_i \in S_{-3:-1} \\ 2 & \text{otherwise} \end{cases} \tag{7}$$

where $S_{1:3}$ denotes the set containing the first three sentences of a document while $S_{-3:-1}$ the set constituted by the last three sentences. Equation (7) holds for our experiment because all the documents used have more than six sentences. The three integers are mapped to a vector space $\mathbf{h}_{\text{sp}} \in \mathbb{R}^{k'}$. The mapping procedure is done like that of word embedding by (2), except that the position embedding matrices are initialized randomly. The concatenation of sentence representation vector $\mathbf{h}$ and sentence position embedding $\mathbf{h}_{\text{sp}}$ formulates the sentence embedding $\mathbf{h}_s = [\mathbf{h}, \mathbf{h}_{\text{sp}}] \in \mathbb{R}^{k+k'}$ which is used as the input of the fully-connected layer.

To avoid overfitting, dropout with a masking probability p is applied on the penultimate layer. Thus, the significance of the sentence is calculated through a regression process governed by

$$\widehat{y} = \sigma(\mathbf{w}_r(\mathbf{h}_s \otimes \mathbf{r}) + b_r) \tag{8}$$

where $\sigma(\cdot)$ is a sigmoid function, $\otimes$ is an element-wise multiplication operator, and $\mathbf{r}$ is the masking vector with $p = 0.5$ in this paper. In addition, a $L2-$norm constraint of the filter weights $\mathbf{w}_r$ is imposed during training as well. The model parameters including word vectors and sentence position embeddings are all fine-tuned via stochastic gradient descent using the Adadelta update rule, which has been shown as an effective and efficient backpropagation algorithm.

B. Multiview Convolutional Neural Networks for Multidocument Summarization

The proposed CNN is able to efficiently extract features of sentences and perform sentence ranking jointly. However, we believe that using a single CNN can hardly capture all the information contained in sentences sufficiently. In practical cases, it is common that human beings may hold significantly different opinions toward the same sentences in summarizing documents. This is because that they own unique perceptions and biases as well as different background knowledge and intentions. Therefore, we leverage on the idea of multiview learning to consider different perspectives toward the same set of documents. The framework of the newly proposed MV-CNN model is depicted in Fig. 2. The success of multiview learning approaches heavily depends on two significant principles, namely, complementary and consensus principles. The former principle demonstrates that complementary information underlying multiview data can be exploited to comprehensively and accurately describe the data and thus improve the learning performance while the latter aims at maximizing the agreement of distinct learners trained on multiple views.

1) Complementary Principle: As foreshadowed in Section II, our new MV-CNN model can be regarded as a multiple-kernel-learning method, where each CNN functions

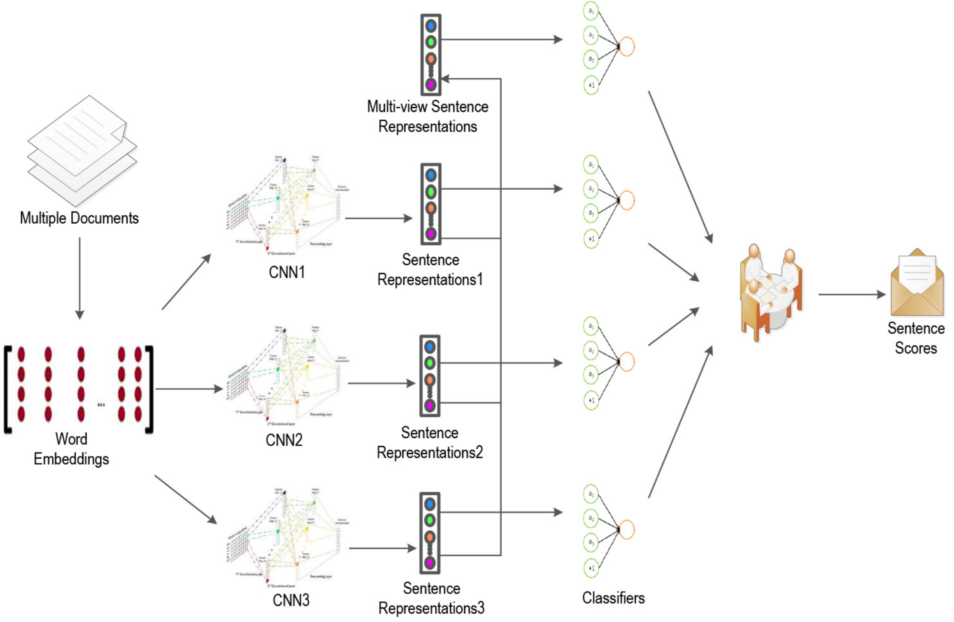

Multi-view Sentence Representations
Multiple Documents
CNN1
Sentence Representations1
Word Embeddings
CNN2
Sentence Representations2
CNN3
Sentence Representations3
Classifiers
Sentence Scores

Fig. 2. MV-CNN framework.

like a kernel. Separate kernels naturally correspond to various views, assessing data from distinct perspectives. The complementary information underlying various views of data help comprehensively evaluate the data. As shown in Fig. 2, our proposed MV-CNN uses multiple CNNs to represent an input sentence with a vector. The CNN in Fig. 2 is a little bit different from the basic CNN structure depicted in Fig. 1. It does not include the fully-connected layer because CNNs are exploited to obtain vector representation of sentences. The fully-connected layers are separately shown as classifiers. The CNNs have distinct window-size filters convolving with the input data, extracting different latent information. We are not concerned with which CNN has the best performance. Instead, we combine the results of distinct CNNs appropriately using the complementary principle. In addition, we also concatenate sentence representations acquired by each CNN into *multiview sentence representations* and use an additional classifier to analyze the new representations. Therefore, complementary principles are applied to our new model at two levels jointly, namely, final stage combining distinct outputs and the intermediate stage combining distinct sentence representations. The final-stage combination is like different summarizers negotiate and compromise to determine the final summaries after getting their own summaries. And the intermediate-stage combination can be thought of as different summarizers exchange one another's opinions first and perform the summarization task together. Therefore, the learning process of the MV-CNN is imitating the human summarizers' behavior.

We formulate a theorem in order to prove that the complementary principle is helpful in improving the learning performance.

Theorem 1: We denote the input as $\mathbf{x}$, real output as y, and the predicted output by a single CNN as $f(\mathbf{x})$. The multiview CNN predicted output is weighted sum of the distinct CNNs' outputs. Then the expected mean squared error of a single CNN is equal or greater than that of MV-CNN.

Proof: As all the CNNs are from the same distribution (denoted as χ), the MV-CNN predicted output is given by

$$f_{\mathrm{MV}}(\mathbf{x}) = E_{f\sim\chi}(f(\mathbf{x})) \tag{9}$$

where the expression $f \sim \chi$ means f conforms to the distribution χ. The expected error of the MV-CNN is given by

$$e_{\mathrm{MV}} = E_x(y - f_{\mathrm{MV}}(\mathbf{x}))^2. \tag{10}$$

The expected error of a single CNN is calculated as follows:

$$e = E_x(y - f(\mathbf{x}))^2. \tag{11}$$

Now, we prove that the MV-CNN can obtain better learning performance than a single CNN if $e \geq e_{\mathrm{MV}}$ holds. We have

$$\begin{aligned} e &= E_x(y - f(\mathbf{x}))^2 \\ &= E_{f\sim\chi}E_x(y - f(\mathbf{x}))^2 \\ &= E_{f\sim\chi}E_x\left(y^2 - 2yf(\mathbf{x}) + f^2(\mathbf{x})\right) \\ &= E_x\left(y^2 - 2yE_{f\sim\chi}(f(\mathbf{x})) + E_{f\sim\chi}\left(f^2(\mathbf{x})\right)\right) \\ &\geq E_x\left(y^2 - 2yf_{\mathrm{MV}}(\mathbf{x}) + E^2_{f\sim\chi}(f(\mathbf{x}))\right) \\ &= E_x\left(y^2 - 2yf_{\mathrm{MV}}(\mathbf{x}) + f^2_{\mathrm{MV}}(\mathbf{x})\right) \\ &= E_x(y - f_{\mathrm{MV}}(\mathbf{x}))^2 = e_{\mathrm{MV}}. \end{aligned} \tag{12}$$

Thus, the usefulness of the complementary principle is established. ■

2) Consensus Principle: The consensus principle is another key for the success of multiview learning. It ensures the similarity of the predicted results of multiple learners on the same data. Dasgupta *et al.* [56] demonstrated that the probability of two independent hypotheses that disagree on two views imposes an upper bound on the error rate of either hypothesis. This relationship can be described by the following inequality:

$$P(f_1 \neq f_2) \geq \max(P(\text{error}(f_1)), P(\text{error}(f_2))). \tag{13}$$

Therefore, the error rate of each hypothesis decreases if the consensus between two hypotheses is enhanced. This means that we can increase each CNN's learning accuracy thus increase the final learning capability if the disagreement of distinct CNNs' predictions are minimized. In the framework of the MV-CNN as shown in Fig. 2, we not only try to minimize the discrepancy between the multiview prediction output and real target but also minimize the disagreement of distinct CNNs' predictions. The predicted saliency score of the input sentence by each CNN is denoted as $f_i(\mathbf{x})$. The consensus principle is governed by

$$\min \sum_{\mathbf{x}} \sum_{i \neq j} \left(f_i(\mathbf{x}) - f_j(\mathbf{x})\right)^2. \tag{14}$$

As complementary and consensus principles have been introduced, the entire summarization procedure is describe below.

3) Preprocessing: Our proposed method is a supervised approach. However, ready-made salience scores are not available for the training data. Therefore, we first preprocess the documents to obtain a salience score for each sentence. The document sets are given together with their reference summaries. We adopt the widely-accepted automatic summarization evaluation metric, ROUGE, to measure the salience. Rouge assesses the quality of an automatic summary by counting the overlapping units, such as n-gram, common word pairs and longest common subsequences between automatic summary and a set of reference summaries. The n-gram ROUGE metric (ROUGE-N) can be computed as follows:

$$\text{ROUGE} - N = \frac{\sum_{S \in \{\text{RefSum}\}} \sum_{n-\text{gram} \in S} \text{Count}_{\text{match}}(n - \text{gram})}{\sum_{S \in \{\text{RefSum}\}} \sum_{n-\text{gram} \in S} \text{Count}(n - \text{gram})} \tag{15}$$

where n stands for the length of the n-gram, Count($n-$gram) is the number of n-grams in the set of reference summaries and $\text{Count}_{\text{match}}(n - \text{gram})$ refers to the number of n-grams co-occurring in a system-generated summary and the set of reference summaries. As ROUGE-1 (R_1) and ROUGE-2 (R_2) most agree with human judgment among all the ROUGE scores [57], we calculate each sentence's score as follows:

$$y = \alpha R_1(\mathbf{x}) + (1 - \alpha) R_2(\mathbf{x}). \tag{16}$$

We set the coefficient $\alpha = 0.5$ for equal weighting of the two scores. The scores R_1 and R_2 are obtained by comparing each sentence in multiple documents with corresponding reference summary.

4) Training: After the preprocessing procedure, we use the training documents to train our MV-CNN model. The single CNN structure has been demonstrated in Section III-A. Besides the real CNNs, the score generated by *multiview sentence representation* is regarded of being assigned by an pseudo CNN. The objective function in terms of minimizing the cross-entropy is used

$$\mathbb{L} = -\sum_{\mathbf{x}} y \ln\left(\sum_i u_i f_i(\mathbf{x})\right) + (1 - y) \ln\left(1 - \sum_i u_i f_i(\mathbf{x})\right) \tag{17}$$

where u_i is the weight of each saliency score assigned by the corresponding CNN. The score weights are updated during training.

Besides the cross-entropy loss function, we also apply the consensus principle to minimize disagreement between every two classifiers. Thus, by combining (14) and (17), we formulate the final objective function as follows:

$$\min \lambda \sum_{\mathbf{x}} \sum_{i \neq j} \left(f_i(\mathbf{x}) - f_j(\mathbf{x})\right)^2 + \mathbb{L} \tag{18}$$

where the term λ is the parameter regulating the two components, which is chosen using cross-validation. In the experiments, we only use three CNNs due to the computation and time constraints. The learning performance is expected to improve if we use more CNNs.

5) Testing: For the testing documents, all the sentences will be used as the input to the trained CNN model to obtain their saliency scores. Next, sentences in each document set can be ranked according to their salience scores. Since a good summary should be not only informative but also nonredundant, we employ the sentence selection method of [58] to select from the ranked sentences. The selection method queries the sentence with highest salience score and adds it to the summary if the similarity of the sentence with all the sentences already in the summary does not exceed a threshold. This sentence selection procedure is especially necessary for MDS because sentences extracted from different documents in one topic can be very similar. The selection process repeats until the length limit of the final summary is met.

The entire learning algorithm of MV-CNN is summarized as Algorithm 1.

IV. Experimental Studies and Discussion

A. Datasets

The benchmark datasets from the DUCs[1] are used to evaluate our proposed MDS system. The datasets are English news articles. In this paper, DUC2001/2002/2004/2006/2007 datasets are used for evaluating generic MDS tasks. The characteristics of the datasets are given in Table I. The table gives the number of document clusters, total document size, and the up limit of summary length. When evaluating on DUC2001/2002/2004, we use two years of data as training data and one year of data as test data. When evaluating on DUC2006/2007, we use one year of data as training data and the other year's data as test data. The reference summaries for each set of documents are given together with the documents.

[1] http://duc.nist.gov

Algorithm 1 Pseudo-Code for MV-CNN

```
1: Pre-process the documents and reference summaries, obtaining the saliency score for each sentence;
2: Construct word embedding table using pre-trained word vectors;
3: Generate sentence position embedding $h_{sp}$ for each sentence using Eq. (7);
4: for i in [1, K] do
5:     For the $ith$ CNN with window size $m^i$, apply two consecutive convolution operation between $k$ filters and the input sentences using Eq. (3-6);
6:     Apply max-pooling operation to get sentence representations $h^i$;
7:     Concatenate $h^i$ and $h_{sp}$ and apply fully-connected sigmoid classifier to obtain regression scores $f_i$ using Eq. (8);
8: end for
9: Concatenate $K$ sentence representations $h^i (i = 1, \cdots, K)$ and $h_{sp}$ and apply fully-connected sigmoid classifier to obtain regression scores $f_{MV}$;
10: Update parameters of the model using the loss function Eq. (18) with the Adadelta.
```

TABLE I
CHARACTERISTICS OF DATA SETS

Year	Clusters	Documents	Length Limit
2001	30	303	100 words
2002	59	533	100 words
2004	50	500	665 bytes
2006	50	1250	250 words
2007	45	1125	250 words

B. Evaluation Metrics

For the evaluation of summarization performance, we employ the widely used ROUGE[2] toolkit [59]. ROUGE has become the standard evaluation metric for DUC since 2004. ROUGE-1 (unigram) and ROUGE-2 (bigram) are used as measure metrics because they most agree with human judgment [60]. Rouge-SU4 is also a very popular metric because it conveys the readability of a candidate summary. Rouge-S is a co-occurrence statistic measuring the overlap of skip-bigrams between a candidate summary and a set of reference summaries. Skip-bigram is any pair of words in their sentence order. Rouge-SU is an extension of Rouge-S which adds unigram as counting unit as well to decrease the chances of zero scores, where there is no skip-bigram overlap. Rouge-SU4 limits that word pairs at most four words apart can form skip-bigrams. The calculation formula of Rouge-SU4 can be easily obtained by replacing the n-gram of (15) by skip-bigram. Therefore, ROUGE-1, ROUGE-2, and Rouge-SU4 scores are reported in our evaluation study. We set the length parameter “-l 100” for DUC2001/2002, “-b 665” for DUC2004, and “-l 250” for DUC2006/2007. Rouge generates precision, recall, and F-measure metrics. We employ F-measure, which is a combination of precision and recall metrics, in our experiment to compare the performances of different algorithms.

C. Parameter Settings

The embeddings of words used in our experiments are initialized using skip-gram neural networks method proposed by Mikolov *et al.* [15] which is available in *word2vec* tool.[3] The model is trained on a part of Google News dataset which has about 100 billion words. The dimension of each word vector is 300. The word vectors are already available and can be directly downloaded from the *word2vec* tool website. The number of filters k and the dimension of sentence position embedding k' are both set to 100. The further increase of the number of filters does not enhance the learning performance obviously but increases the complexity of the model. The threshold value of sentence selection for MDS is set to 0.5. These parameters are determined using cross validation. Only three CNNs are used in our experiment because of the computation and time constraints. We expect to obtain better learning performance when using more CNNs. In order to have a fair comparison with the baseline CNN, the filter window sizes for the three CNNs are chosen as 3–5. Other hyper-parameters are fine tuned via the stochastic gradient descent method Adadelta. All the simulation studies are conducted using an NVIDIA Tesla K20c GPU on a Windows Server with 2.0 GHZ CPU and 256GB RAM.

D. Experiment Results

In order to demonstrate the summarization performance of our model, we compare it with several state-of-the-art extractive summarization techniques. The most baseline method is Random which selects sentences randomly from the candidate set. As mentioned in the related work section, extractive summarization methods are roughly classified into three categories and our method belongs to the supervised category. We compare with methods from the other two categories. Lead is a representative of the methods based on sentence positions and article structure while LSA belongs to the unsupervised category. Two most recently published results, namely, MDS-Sparse [61] and RA-MDS [62] are also included for comparison. Both methods are developed based on sparse coding and belong to unsupervised category. We directly retrieve the results of LSA, MDS-Sparse, and RA-MDS in [61] and [62] because the datasets and evaluation metrics are standard. As the two papers only did experiments on DUC2006/2007, the results for the three methods on DUC2001/2002/2004 are not given in our experiments. These comparison systems are basic baselines.

To demonstrate the effectiveness of incorporating multiview learning into CNN, we compare our method with two other CNN baselines, i.e., BasicCNN and PriorSum. BasicCNN is the basic CNN used in our model with a fixed window size. PriorSum is the model proposed by Cao *et al.* [4]. In order to have a fair comparison, we only concatenate position features to the sentence features obtained by PriorSum excluding the other handcrafted features. We experiment on BasicCNN using

[2] ROUGE-1.5.5 with options: -n 2 -x -m -2 4 -u -c 95 -r 1000 -f A -p 0.5 -t 0 -d.

[3] https://code.google.com/p/word2vec

TABLE II
ROUGE-1 SCORE COMPARISON RESULTS (%)

System	DUC01	DUC02	DUC04	DUC06	DUC07
Random	28.25	27.78	31.47	28.05	30.20
Lead	29.43	28.68	32.10	30.76	31.19
LSA	-	-	-	24.42	25.98
MDS-Sparse	-	-	-	34.44	35.40
RA-MDS[4]	-	-	-	**39.1**	40.3
BasicCNN	35.38	36.07	38.24	37.98	40.16
PriorSum	35.76	36.24	38.58	38.16	40.48
MV-CNN	**35.99**	**36.71**	**39.07**	**38.65**	**40.92**

TABLE III
ROUGE-2 SCORE COMPARISON RESULTS (%)

System	DUC01	DUC02	DUC04	DUC06	DUC07
Random	4.23	4.76	4.97	4.61	4.62
Lead	4.03	5.28	6.38	4.84	5.76
LSA	-	-	-	3.02	4.06
MDS-Sparse	-	-	-	5.12	6.45
RA-MDS	-	-	-	**8.1**	**9.2**
BasicCNN	7.69	8.73	9.80	7.55	8.60
PriorSum	7.79	8.85	9.86	7.58	8.77
MV-CNN	**7.91**	**9.02**	**10.06**	**7.91**	**9.11**

TABLE IV
ROUGE-SU4 SCORE COMPARISON RESULTS (%)

System	DUC01	DUC02	DUC04	DUC06	DUC07
Random	8.46	8.61	8.82	8.79	8.76
Lead	8.88	9.53	10.23	8.65	10.20
LSA	-	-	-	7.10	8.34
MDS-Sparse	-	-	-	10.72	11.67
RA-MDS	-	-	-	13.6	14.6
BasicCNN	12,79	14.22	12.96	13.45	14.98
PriorSum	12.84	14.37	13.05	13.59	14.97
MV-CNN	**13.16**	**14.92**	**13.67**	**14.09**	**15.34**

three window sizes ($m = 3, 4, 5$) and show the best result. The other settings of the two CNN models are the same with the MV-CNN. The three CNN models fall in the category of supervised methods.

Tables II–IV are the overall performance comparison results. It is clear that the MV-CNN outperforms all the other algorithms except the RA-MDS. Among all the algorithms, the LSA gives the poorest performance on DUC2006/2007. The LSA exploits SVD on term frequency matrix and assumes that sentences with highest eigenvalues are most informative sentences in a document corpus. Our experiment results suggest that such hypothesis may not be true for human understanding. The table also shows that the Lead performs slightly better than the random method. One reason could be because that the authors like to put summary sentences at the beginning of the documents. The two sparse coding-based methods achieve state-of-the-art performance on DUC2006/2007 proving that sparse coding helps to find informative sentences out of documents. The RA-MDS achieves the best performance for Rouge-1 and Rouge-2 on DUC2006. However, it must be highlighted that the RA-MDS exploits a set of user comments associated with the documents to help with the summarization task. This means that it has more input information which is pretty unfair for the other compared methods. Irrespective of this, the MV-CNN shows higher Rouge-SU4 score on DUC2006 and Rouge-1 and Rouge-SU4 scores on DUC2007 than RA-MDS. This effectively demonstrates the outstanding summarization ability of the MV-CNN. Besides, one of the biggest advantages of our method is that it does not need hand-crafted features. The pretrained word embedding has saved us an enormous amount of time and effort, enabling us to avoid the intensive human labor of feature engineering. The MV-CNN outperforms the other two CNN baselines on all metrics on all the datasets. This proves the effectiveness of the incorporation of multiview learning idea. It should be noted that MV-CNN obtains much better Rouge-SU4 scores than all the compared methods. This indicates that the MV-CNN has good readability which may because that multiview learning excludes bias of a single CNN and assigns highest score to the most appropriate sentence from a similar sentence list.

Some researchers may argue that deep learning methods demand prohibitive computational cost and memory because there are so many parameters to tune and store in the memory. However, the problem is no longer a limitation with the progress in hardware, software, algorithm parallelization, and the efficient use of graphics processing units. It takes less than 2 h to run 30 iterations for the MV-CNN in our experiment. It must be highlighted that conventional machine learning methods require good feature extractors before applying learning models. If we take the entire pipeline into consideration, deep learning methods should use much less time and efforts. On the other hand, the complexity of MV-CNN increases compared with the standard CNN because two convolution layers and multiple CNNs are used. But we believe it is worthy of increasing a little complexity to enhance the learning performance.

We perform paired t-test between all three Rouge scores of MV-CNN and those of other approaches in order to verify that MV-CNN achieves significantly better performance compared with other approaches. Paired t-test is a statistical technique that is used to compare population means of two approaches. It can be used for comparison of two different methods of measurement when the measurements are applied to the same subjects. For each approach, we run the experiment ten times for statistical comparison (the results of the LSA, MDS-Sparse, and RA-MDS are supposed to be the same for ten times). We calculate the differences of each time for each pair of methods. Then the mean difference and standard deviation of differences are calculated in order to obtain the t-statistic. Comparing this value with the t-distribution table gives the p-value for the paired t-test. The p-value is used to indicate whether the measurements of the two methods are statistically different. The RA-MDS is excluded for paired t-test for fairness. We find the associated p-values for all metrics on all the datasets are all smaller than 0.0001. The associated p-values of the paired t-test between MV-CNN and other approaches except for the two CNN baselines are even smaller than 1×10^{-10}. Therefore, we can conclude that our proposed method obtains superior performance compared with the other methods at nearly 100% confidence.

E. Discussion

The comparison results demonstrate that multiview learning idea is highly effective in improving learning capability. As foreshadowed in Section III-B, complementary and consensus

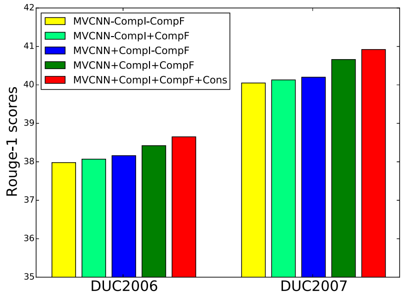


Fig. 3. Rouge-1 scores (%) comparison by impact of complementary and consensus principles.

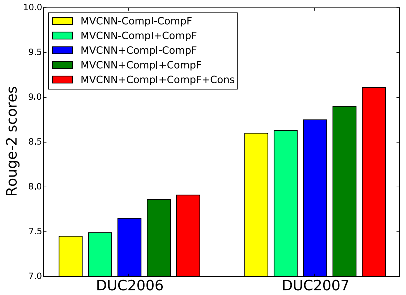


Fig. 4. Rouge-2 scores (%) comparison by impact of complementary and consensus principles.

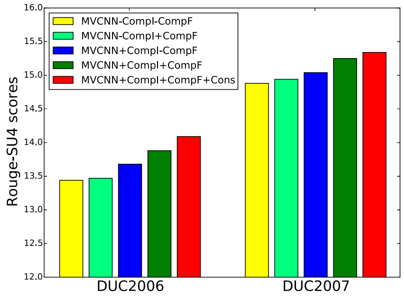


Fig. 5. Rouge-SU4 scores (%) comparison by impact of complementary and consensus principles.

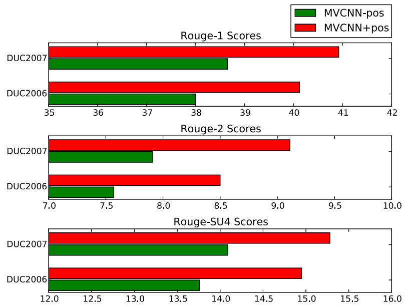


Fig. 6. Rouge scores (%) comparison by impact of sentence position embedding. From top to bottom are Rouge-1, Rouge-2, and Rouge-SU4, successively.

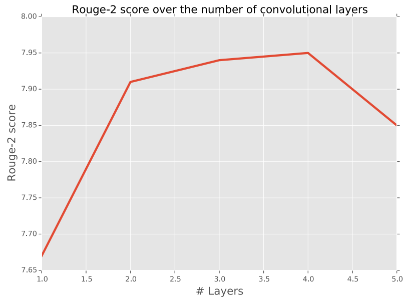


Fig. 7. Rouge-2 scores (%) on DUC2006 by effect of the number of convolutional layers.

principles are the keys to the success of multiview learning. The complementary principle shows that multiple views describing the sentences from distinct perspectives can help comprehensively extract underlying information. The consensus principle leads to the different *summarizers* reaching a maximal consensus. In this section, we analyze the impacts of the two principles using experiments on DUC2006/2007. In addition, our model adopts a novel component, i.e., sentence position embedding. The impact of sentence position embedding will be demonstrated in this section as well.

1) Impact of the Complementary and Consensus Principles: The MV-CNN applies complementary principles at two levels, namely, final stage combining distinct outputs and the intermediate stage combining distinct sentence representations. We analyze the impact of each level complementary setting by excluding the final-stage combination and intermediate combination, respectively, from the MV-CNN model. Similarly, the impact of the consensus principle is demonstrated by comparing the MV-CNN with and without consensus setting.

The comparison results are shown in Figs. 3–5. In the figures, MVCNN+CompI+CompF+Cons denotes the complete MV-CNN model, MVCNN+CompI+CompF denotes the MV-CNN model without consensus setting but with a full complementary setting, and MVCNN-CompI+CompF and MVCNN+CompI-CompF denote the MV-CNN without intermediate complementary and final complementary setting, respectively. Furthermore, MVCNN-CompI-CompF denotes CNN without any complementary and consensus settings, which is actually the BasicCNN. We ignore the consensus principle when analyzing the impact of the complementary principle. It can be seen from the figures that both complementary and consensus principles are useful in performance improvement. Comparing the green and blue bars, we can conclude that the intermediate complementary setting is more important than the final-stage complementary. Employing both stage complementary further enhances the model's learning capability (blue bar). These experiments demonstrate the effectiveness and efficiency of multiview learning empirically.

2) Impact of the Sentence Position Embedding: The position a sentence occurred in one document can be critical for determining how important the sentence is for the document. A position embedding instead of a position integer is used because the effect of an integer feature may almost be ignored when it is concatenated with the long sentence vector.

Comparison results are shown in Fig. 6. In the figure, MVCNN+pos denotes the complete MV-CNN model and

TABLE V
ONE MANUAL SUMMARY AND AN AUTOMATIC SUMMARY GENERATED BY THE PROPOSED MODEL ON THE TOPIC "AN AIR FRANCE CONCORDE CRASH"

Reference Summary[5]	***The only supersonic passenger plane–the Concorde–crashed on July 25, 2000, killing all 109 on board and 4 on the ground.*** *......* ***The plane had 100 seats and flew 1,300 mph at 60,000 feet. It took less that 4 hours to cross the Atlantic*** *carrying the rich, the famous, and business people. ......* ***The crash was caused by a piece of metal falling off of a Continental DC10 that used the runway before the Concorde****. The metal cut the Concorde's tire and flying rubber damaged the engines* ***causing a fuel leak that burst into flame****, too late to abort. ......* ***Almost all of the passengers were Germans going to New York.*** *Air France paid each family $20,000 to cover immediate expenses.* ***Memorial services were held in Paris, in Cologne, Germany, and later at the crash site. Air France immediately grounded its fleet and British Airways stopped flights on August 15****. ......*
Automatic Summary	***An Air France Concorde en route to New York crashes outside Paris shortly after takeoff, killing all 109 people on board and four people on the ground. Most of the crash victims were Germans flying to New York*** *to join a luxury cruise through the Caribbean. ...... After the crash,* ***Air France agreed to temporarily ground its five remaining Concordes, and British Airways grounded its two remaining flights for Tuesday night****. ...... French investigators have said they believe that* ***a thin strip of metal fell off the Continental plane, which had taken off on the same runway as the Concorde****, and set off the events that* ***caused the Air France jet to burst into flames*** *and crash within minutes after its takeoff. ...... One month after an Air France Concorde crashed outside Paris,* ***victims' relatives gathered Saturday to remember their loved ones near the site where the supersonic jet went down.*** *......* ***The sleek, needle-nosed aircraft could cross the Atlantic at an altitude of 60,000 feet and at 1,350 mph****, completing the trip from London to New York in less than four hours – half the time of regular jets. ......*

MVCNN-pos denotes MV-CNN model without using sentence position embedding. It can be easily concluded from the figure that the existence of sentence position embedding helps enhance the performance on all the three metrics on both DUC2006 and DUC2007 datasets. This is because position information of a sentence in one document can be significant to determine whether it should be included into the final summary.

F. Effect of Convolutional Layer Number

The basic CNN used in our framework uses two convolutional layers. Empirical results are given in this section to justify the statement. We show the Rouge-2 scores experimented on DUC2006 dataset over the number of convolutional layers in Fig. 7. The figure demonstrates that two convolutional layers achieve better learning performance than one convolutional layer model. However, the performance only increases marginally and even decrease because of over-fitting with the increasing number of convolutional layers. Similar trends are detected on other metrics experimented on other datasets. Besides, the computation complexity increases fast if using more layers. Therefore, two convolutional layers are used in our model.

G. Case Study

A real case is shown in this section. The 250-word automatic summary generated by the proposed model for the document cluster D0631 of DUC2006 is given in Table V. One manual reference summary is given for comparison. All the 25 articles in this cluster are over the same topic "An Air France Concorde crash." There are over 700 sentences in this document set, while the reference summary and automatic summary each contains about 15 sentences. From the table, we can see that the sentences in bold almost convey the same meaning. The real case study proves that our proposed method can extract the most informative sentences.

V. CONCLUSION

In this paper, an enhanced CNNs termed MV-CNNs has been successfully developed to obtain the features of sentences and rank sentences jointly. We leverage on pretrained word embedding to represent sentences so as to avoid the intensive labor of feature engineering. This makes our model much easier to be implemented. The biggest innovation of the new model is the incorporation of multiview learning into standard CNN. The two-level complementary and consensus principles of multiview learning are fully exploited to improve the model's learning performance. To the best of our knowledge, this is the first time that the combination of multiview learning and CNN is used for MDS. Novel sentence position embedding is introduced to help retrieve information in the documents, advancing the learning ability of proposed model to a higher level. Although our CNN model incurs higher computational complexity and memory demand compared with conventional machine learning algorithms, it has a feature learning capability, which outweighs the possible increase of complexity.

The learning process of the MV-CNN imitates the human summarizers' behavior. Experiment results demonstrate that our model performs remarkably well, achieving better performance compared with state-of-the-art approaches. This paper can only be applied to extractive summarization tasks. However, the generated summaries may not be so readable as human-generated summaries. Our future work will attempt to improve our model and apply to abstractive summarization tasks.

REFERENCES

[1] A. F. T. Martins and N. A. Smith, "Summarization with a joint model for sentence extraction and compression," in *Proc. Workshop Integer Linear Program. Nat. Lang. Process.*, Leuven, Belgium, 2009, pp. 1–9.

[2] K.-Y. Chen *et al.*, "A recurrent neural network language modeling framework for extractive speech summarization," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Chengdu, China, 2014, pp. 1–6.

[3] M. Denil, A. Demiraj, N. Kalchbrenner, P. Blunsom, and N. de Freitas, "Modelling, visualising and summarising documents with a single convolutional neural network," Univ. Oxford, Oxford, U.K., Tech. Rep., 2014.

[4] Z. Cao *et al.*, "Learning summary prior representation for extractive summarization," in *Proc. 53rd Annu. Meeting Assoc. Comput. Linguist.*, Beijing, China, 2015, pp. 829–833.

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[6] G. Hinton *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 82–97, Nov. 2012.

[7] K. Zeng, J. Yu, R. Wang, C. Li, and D. Tao, "Coupled deep autoencoder for single image super-resolution," *IEEE Trans. Cybern.*, vol. PP, no. 99, pp. 1–11, 2016, doi: 10.1109/TCYB.2015.2501373.

[8] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[9] Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin, "A neural probabilistic language model," *J. Mach. Learn. Res.*, vol. 3, pp. 1137–1155, Feb. 2003.

[10] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 3111–3119.

[11] Y. Bengio, H. Schwenk, J.-S. Senécal, F. Morin, and J.-L. Gauvain, "Neural probabilistic language models," in *Innovations in Machine Learning*. Heidelberg, Germany: Springer, 2006, pp. 137–186.

[12] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proc. 25th Int. Conf. Mach. Learn.*, Helsinki, Finland, 2008, pp. 160–167.

[13] A. Mnih and G. E. Hinton, "A scalable hierarchical distributed language model," in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, 2009, pp. 1081–1088.

[14] J. Turian, L. Ratinov, and Y. Bengio, "Word representations: A simple and general method for semi-supervised learning," in *Proc. 48th Annu. Meeting Assoc. Comput. Linguist.*, Uppsala, Sweden, 2010, pp. 384–394.

[15] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, Scottsdale, AZ, USA, 2013.

[16] R. Socher, C. C.-Y. Lin, C. D. Manning, and A. Y. Ng, "Parsing natural scenes and natural language with recursive neural networks," in *Proc. 28th Int. Conf. Mach. Learn. (ICML)*, Bellevue, WA, USA, 2011, pp. 129–136.

[17] E. H. Huang, R. Socher, C. D. Manning, and A. Y. Ng, "Improving word representations via global context and multiple word prototypes," in *Proc. 50th Annu. Meeting Assoc. Comput. Linguist. Long Papers*, vol. 1. 2012, pp. 873–882.

[18] W. Y. Zou, R. Socher, D. M. Cer, and C. D. Manning, "Bilingual word embeddings for phrase-based machine translation," in *Proc. EMNLP*, Seattle, WA, USA, 2013, pp. 1393–1398.

[19] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[20] L. Zhang and P. N. Suganthan, "Visual tracking with convolutional random vector functional link network," *IEEE Trans. Cybern.*, vol. PP, no. 99, pp. 1–11, 2016, doi: 10.1109/TCYB.2016.2588526.

[21] B. Du *et al.*, "Stacked convolutional denoising auto-encoders for feature representation," *IEEE Trans. Cybern.*, vol. PP, no. 99, pp. 1–11, 2016, doi: 10.1109/TCYB.2016.2536638.

[22] Y. Wei *et al.*, "Cross-modal retrieval with CNN visual features: A new baseline," *IEEE Trans. Cybern.*, vol. PP, no. 99, pp. 1–12, 2016, doi: 10.1109/TCYB.2016.2519449.

[23] R. Collobert *et al.*, "Natural language processing (almost) from scratch," *J. Mach. Learn. Res.*, vol. 12, pp. 2493–2537, Aug. 2011.

[24] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, "A convolutional neural network for modelling sentences," in *Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics (ACL)*, Baltimore, MD, USA, 2014, pp. 655–665.

[25] M. J. Er, Y. Zhang, N. Wang, and M. Pratama, "Attention pooling-based convolutional neural network for sentence modelling," *Inf. Sci.*, vol. 373, pp. 388–403, Dec. 2016.

[26] J. Weston, S. Chopra, and K. Adams, "♯TAGSPACE: Semantic embeddings from hashtags," in *Proc. Conf. Empir. Methods Nat. Lang. Process. (EMNLP)*, Doha, Qatar, 2014, pp. 1822–1827.

[27] Y. Kim, "Convolutional neural networks for sentence classification," in *Proc. Conf. Empirical Methods Nat. Lang. Process. (EMNLP)*, Doha, Qatar, 2014, pp. 1746–1751.

[28] W.-T. Yih, X. He, and C. Meek, "Semantic parsing for single-relation question answering," in *Proc. ACL*, Baltimore, MD, USA, 2014, pp. 643–648.

[29] D. Zeng, K. Liu, S. Lai, G. Zhou, and J. Zhao, "Relation classification via convolutional deep neural network," in *Proc. COLING*, Dublin, Ireland, 2014, pp. 2335–2344.

[30] C. Xu, D. Tao, and C. Xu, "A survey on multi-view learning," *Neural Comput. Appl.*, vol. 23, nos. 7–8, pp. 2031–2038, 2013.

[31] I. Mani and M. T. Maybury, *Advances in Automatic Text Summarization*, vol. 293. Cambridge, MA, USA: MIT Press, 1999.

[32] M. Wasson, "Using leading text for news summaries: Evaluation results and implications for commercial summarization applications," in *Proc. 36th Annu. Meeting Assoc. Comput. Linguist. 17th Int. Conf. Comput. Linguist.*, vol. 2. Montreal, QC, Canada, 1998, pp. 1364–1368.

[33] Y. Gong and X. Liu, "Generic text summarization using relevance measure and latent semantic analysis," in *Proc. 24th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, New Orleans, LA, USA, 2001, pp. 19–25.

[34] G. Erkan and D. R. Radev, "LexRank: Graph-based lexical centrality as salience in text summarization," *J. Artif. Intell. Res.*, vol. 22, no. 1, pp. 457–479, 2004.

[35] R. Mihalcea and P. Tarau, *TextRank: Bringing Order Into Texts*. Stroudsburg, PA, USA: Assoc. Comput. Linguist., 2004, pp. 404–411.

[36] J. Carbonell and J. Goldstein, "The use of MMR, diversity-based reranking for reordering documents and producing summaries," in *Proc. 21st Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, Melbourne, VIC, Australia, 1998, pp. 335–336.

[37] X. Wan and J. Yang, "Multi-document summarization using cluster-based link analysis," in *Proc. 31st Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, Singapore, 2008, pp. 299–306.

[38] H. Lin and J. Bilmes, "A class of submodular functions for document summarization," in *Proc. 49th Annu. Meeting Assoc. Comput. Linguist. Human Lang. Technol.*, vol. 1. Portland, OR, USA, 2011, pp. 510–520.

[39] M. Galley, "A skip-chain conditional random field for ranking meeting utterances by importance," in *Proc. Conf. Empir. Methods Nat. Lang. Process.*, Sydney, NSW, Australia, 2006, pp. 364–372.

[40] C. Li, X. Qian, and Y. Liu, "Using supervised bigram-based ILP for extractive summarization," in *Proc. ACL*, Sofia, Bulgaria, 2013, pp. 1004–1013.

[41] K. Hong and A. Nenkova, "Improving the estimation of word importance for news multi-document summarization," in *Proc. EACL*, Gothenburg, Sweden, 2014, pp. 712–721.

[42] Y. Hu and X. Wan, "PPSGen: Learning to generate presentation slides for academic papers," in *Proc. 23rd Int. Joint Conf. Artif. Intell.*, Beijing, China, 2013, pp. 2099–2105.

[43] Z. Cao, F. Wei, L. Dong, S. Li, and M. Zhou, "Ranking with recursive neural networks and its application to multi-document summarization," in *Proc. AAAI Conf. Artif. Intell.*, Austin, TX, USA, 2015, pp. 2153–2159.

[44] G. PadmaPriya and K. Duraiswamy, "An approach for text summarization using deep learning algorithm," *J. Comput. Sci.*, vol. 10, no. 1, pp. 1–9, 2014.

[45] Q. V. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proc. 31st Int. Conf. Mach. Learn. (ICML)*, Beijing, China, 2014, pp. 1188–1196.

[46] M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus, "Deconvolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, San Francisco, CA, USA, 2010, pp. 2528–2535.

[47] W. Wang and Z.-H. Zhou, "A new analysis of co-training," in *Proc. 27th Int. Conf. Mach. Learn. (ICML)*, Haifa, Israel, 2010, pp. 1135–1142.

[48] S. Yu, B. Krishnapuram, R. Rosales, and R. B. Rao, "Bayesian co-training," *J. Mach. Learn. Res.*, vol. 12, pp. 2649–2680, Sep. 2011.

[49] M. Szafranski, Y. Grandvalet, and A. Rakotomamonjy, "Composite kernel learning," *Mach. Learn.*, vol. 79, no. 1, pp. 73–103, 2010.

[50] M. Gönen and E. Alpaydın, "Multiple kernel learning algorithms," *J. Mach. Learn. Res.*, vol. 12, pp. 2211–2268, Jul. 2011.

[51] N. Quadrianto and C. H. Lampert, "Learning multi-view neighborhood preserving projections," in *Proc. 28th Int. Conf. Mach. Learn. (ICML)*, Bellevue, WA, USA, 2011, pp. 425–432.

[52] D. Zhai, H. Chang, S. Shan, X. Chen, and W. Gao, "Multiview metric learning with global consistency and local smoothness," *ACM Trans. Intell. Syst. Technol.*, vol. 3, no. 3, p. 53, 2012.

[53] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller, "Multi-view convolutional neural networks for 3D shape recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, Santiago, Chile, 2015, pp. 945–953.

[54] H. Lin *et al.*, "User-level psychological stress detection from social media using deep neural network," in *Proc. ACM Int. Conf. Multimedia*, Orlando, FL, USA, 2014, pp. 507–516.

[55] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. 30th Int. Conf. Mach. Learn.*, vol. 30. Atlanta, GA, USA, 2013.

[56] S. Dasgupta, M. L. Littman, and D. McAllester, "PAC generalization bounds for co-training," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 1. Vancouver, BC, Canada, pp. 375–382, 2001.

[57] C.-Y. Lin and E. Hovy, "Automatic evaluation of summaries using N-gram co-occurrence statistics," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguist. Human Lang. Technol.*, vol. 1. Edmonton, AB, Canada, 2003, pp. 71–78.

[58] Y. Li and S. Li, "Query-focused multi-document summarization: Combining a topic model with graph-based semi-supervised learning," in *Proc. COLING 25th Int. Conf. Comput. Linguist. Tech. Papers*, Dublin, Ireland, 2014, pp. 1197–1207.

[59] C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in *Proc. Workshop Text Summarization Branches Out (ACL)*, Barcelona, Spain, 2004, pp. 74–81.

[60] K. Owczarzak, J. M. Conroy, H. T. Dang, and A. Nenkova, "An assessment of the accuracy of automatic evaluation in summarization," in *Proc. Workshop Eval. Metrics Syst. Comparison Autom. Summarization*, Montreal, QC, Canada, 2012, pp. 1–9.

[61] H. Liu, H. Yu, and Z.-H. Deng, "Multi-document summarization based on two-level sparse representation model," in *Proc. 29th AAAI Conf. Artif. Intell.*, Austin, TX, USA, 2015, pp. 196–202.

[62] P. Li, L. Bing, W. Lam, H. Li, and Y. Liao, "Reader-aware multi-document summarization via sparse coding," in *Proc. 24th Int. Joint Conf. Artif. Intell. (IJCAI)*, Buenos Aires, Argentina, 2015, pp. 1270–1276.

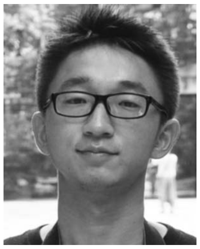
Yong Zhang (S'15) received the B.Eng. degree from Zhejiang University, Hangzhou, China, in 2013. He is currently pursuing the Ph.D. degree with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore.

His current research interests include machine learning, natural language processing, and computational intelligence.

Meng Joo Er (SM'06) received the M.Eng. and Ph.D. degrees from the National University of Singapore, Singapore, and the Australian National University, Australia, in 1988 and 1992, respectively.

He is currently a Full Professor with Electrical and Electronic Engineering, Nanyang Technological University, Singapore. He has authored five books, 16 book chapters, and over 500 refereed journal and conference papers in the below mentioned areas. His current research interests include intelligent control theory and applications, computational intelligence, robotics and automation, sensor networks, biomedical engineering, and cognitive science.

Prof. Er was a recipient of the Institution of Engineers, Singapore (IES) Prestigious Engineering Achievement Award in 2011 and 2015, for the significant and impactful contributions to Singapore's development by his research works, the only dual winner in Singapore IES Prestigious Publication Award in Application in 1996, IES Prestigious Publication Award in Theory in 2001, the Teacher of the Year Award for the School of EEE in 1999, the School of EEE Year 2 Teaching Excellence Award in 2008, the Most Zealous Professor of the Year Award 2009, the Outstanding Mentor Award 2014, the Best Session Presentation Award at the World Congress on Computational Intelligence in 2006, the Best Presentation Award at the International Symposium on Extreme Learning Machine 2012, the IEEE Outstanding Volunteer Award (Singapore Section), the IES Silver Medal in 2011 in recognition of his outstanding contributions to professional bodies, and over 50 awards at international and local competitions. Under his leadership as the Chairman of the IEEE CIS Singapore Chapter from 2009 to 2011, the Singapore Chapter won the CIS Outstanding Chapter Award 2012. He currently serves as an Editor-in-Chief of two international journals, namely *Transactions on Machine Learning and Artificial Intelligence* and the *International Journal of Electrical and Electronic Engineering and Telecommunications*, an Area Editor of the *International Journal of Intelligent Systems Science*, an Associate Editor of thirteen refereed international journals, including the IEEE TRANSACTION ON FUZZY SYSTEMS and the IEEE TRANSACTIONS ON CYBERNETICS, as well as an Editorial Board Member of the EE Times. He is a highly sought-after speaker and he has been invited to deliver over 60 keynote speeches and invited talks overseas. Due to outstanding achievements in research and education, he is listed Who's Who in Engineering Singapore, Second Edition, 2013.



Rui Zhao received the B.Eng. degree in measurement and control from Southeast University, Nanjing, China, in 2012. He is currently pursuing the Ph.D. degree from the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore.

His current research interests include text mining and machine learning.

Mahardhika Pratama (M'14) received the Ph.D. degree from the University of New South Wales (UNSW), Sydney, NSW, Australia, in 2014, with a special approval of the UNSW higher degree committee.

He is currently a Lecturer with the Department of Computer Science and IT, La Trobe University, Melbourne, VIC, Australia. He was with the Centre of Quantum Computation and Intelligent System, University of Technology, Sydney, as a Post-Doctoral Research Fellow of Australian Research Council Discovery Project. He has published over 50 high-quality papers in journals and conferences, and has been invited to deliver keynote speeches in international conferences.

Dr. Pratama was a recipient of various competitive research awards in the past 5 years, namely the Institution of Engineers, Singapore Prestigious Engineering Achievement Award in 2011, the UNSW High Impact Publication Award in 2013 and 2014, and the Outstanding Ph.D. Research Achievement Award. He serves as an Editor-in-Chief of the *International Journal of Business Intelligence and Data Mining*.